

WHAT IS CLAIMED IS:

1. In a controller for a document production system, a method for associating electronic data files to particular document components of a document, comprising:

- a) creating a document node as a parent node;
- b) selecting one of a set of document forms to apply to the document;
- c) creating a document component node as a sub-node of the document node; and
- d) associating an electronic data file with the document component node.

2. The method of **Claim 1**, wherein the step of selecting one of a set of document forms comprises selecting one from a set of at least seven document forms.

3. The method of **Claim 1**, further comprising applying to the electronic data file rules determined by selection of the document form.

4. The method of **Claim 3**, further comprising classifying the document component into one of a set of document component types that are permitted under the form rules for the selected document form.

5. The method of **Claim 4**, further comprising verifying that the document component conforms to the form rules for the document component type into which the document component has been classified.

6. The method of **Claim 4**, further comprising prompting a user of the controller to input required attributes for the document component type into which the document component has been classified.

7. The method of **Claim 4**, further comprising displaying a list of each of the document component nodes created as sub-nodes of the document node together with an indication of the document component type into which the contents of each document component node has been classified.

8. The method of **Claim 1**, further comprising prompting a user to classify the document component as one of a set comprising a variable component and a static component.

9. The method of **Claim 8**, further comprising, in response to classifying the document component as a variable component, prompting the user to designate data address information to enable accessing of variable data during processing of the variable component.

10. The method of **Claim 1**, further comprising allowing a user to classify the document component as one of a set comprising an external component and an internal component.

11. The method of **Claim 10**, further comprising, in response to classifying the document component as an external component, prompting the user for an integrity descriptor for the external component.

12. In a controller for a document production system, such controller having an associated display, a method for providing a graphical representation of a job model for a document, comprising:

- a) inputting into the controller an identifier for the document to be represented;
- b) retrieving job model data regarding the identified document from a database;
- c) displaying a graphical view of the job model arranged as a hierarchical tree of the document and a plurality of its document components.

13. The method of **Claim 12**, wherein the step of displaying further comprises using commercially available software with features enabling hierarchical association between different files identified to the software.

14. The method of **Claim 12**, wherein the step of displaying further comprises displaying a representation of at least one document component as a graphic icon.

15. The method of **Claim 12**, wherein the step of displaying further comprises displaying a text description of at least one document component.

16. The method of **Claim 12**, further comprising altering at least one attribute of at least one document component by manipulating the portion of the graphical view that represents such document component.

17. The method of **Claim 16**, wherein the step of altering comprises adding a document component to the job model.

18. The method of **Claim 16**, wherein the controller is used for control of at least one finishing device and wherein the step of altering comprises directing the orientation in which at least one document component is to be loaded into the finishing device.

19. The method of **Claim 12**, wherein the step of displaying further comprises displaying an indicator of the status of at least one document component.

20. The method of **Claim 19**, wherein the document production system comprises a plurality of finishing operations and wherein the step of displaying status comprises displaying an indicator of the status of the document component in relation to at least one finishing operation.

21. The method of **Claim 19**, wherein the step of displaying an indicator of status further comprises displaying an indicator of a fault in the processing of a document component.

22. The method of **Claim 21**, further comprising, in response to an indicator of a fault in the processing of a document component, displaying a system diagnostic form to provide information to a user concerning the fault.

23. The method of **Claim 19**, wherein the step of retrieving comprises:

a) retrieving job model data arranged as a sequence of nodes associated with a plurality of job segments within the document; and

b) using a recursive algorithm to inquire concerning the status of the job segments within the document.

24. The method of **Claim 12**, wherein the document production system comprises a plurality of finishing devices and wherein the step of retrieving comprises retrieving data concerning the status of at least one finishing device.

25. The method of **Claim 24**, wherein the step of retrieving data concerning the status of a finishing device further comprises using a recursive algorithm to identify a plurality of finishing devices from which such data is to be obtained.

26. The method of **Claim 12**, wherein the step of displaying further comprises displaying a representation of at least one job segment as a sub-node of a document component.

27. The method of **Claim 12**, further comprising:

- a) associating a unique identifier for each item displayed within the hierarchical tree; and
- b) manipulating at least one unique identifier by changing its representation on the display.

28. The method of **Claim 12**, wherein the step of displaying the job model as a hierarchical tree further comprises enabling display of an unlimited number of levels within the hierarchical tree.

29. The method of **Claim 12**, wherein:

- a) the production system comprises a plurality of finishing operations;
- b) the document may be finished using a plurality of possible threads of such finishing operations; and
- c) the step of displaying the job model as a hierarchical tree further comprises displaying a plurality of threads using a perspective view of representations of a plurality of finishing devices arranged as threads

30. The method of **Claim 29**, wherein the document is comprised of job segments and wherein the step of displaying further comprises displaying a map showing at least one thread that a job segment may take through the finishing devices.

31. The method of **Claim 29**, wherein the document is comprised of job segments and wherein the step of displaying further comprises displaying data comprised of the quantity of job segments required to complete the job.

32. The method of **Claim 29**, further comprising selecting at least one thread for performing the finishing operations.

33. The method of **Claim 32**, further comprising displaying the status of at least one thread before the step of selecting a thread for performing the finishing operation.

34. The method of **Claim 12**, wherein the step of retrieving job model data comprises using a recursive algorithm wherein each item within the hierarchical tree is associated with a node of information, wherein each node is embedded with an identifying reference to at least one other node within the hierarchical tree, and wherein the recursive algorithm operates by recursively calling itself until all nodes within the hierarchical tree have been identified.

35. In a finishing system having at least one database for storing information concerning the job model for a job, including identification and descriptions of the job and job segments stored as nodes of information within the database, a graphical user interface, comprising:

a) a opening screen with user options to select a database, a job stored in the database, and a type of report for displaying information concerning the job;

b) an optional display a selected job in a hierarchical view with multiple nodes arranged in a tree view showing the relationship of each node to other nodes within the same job; and

c) an optional display of information describing the status of each job segment within the job.